Treball de Fi de Grau

Grau en Enginyeria en Tecnologies Industrials

Utilización del Álgebra lineal para la clasificación por rangos

MEMÒRIA

Autor: Inés Morán Franqués

Director/s: María Isabel García Planas

Convocatòria: Gener 2016

Escola Tècnica Superior
d’Enginyeria Industrial de Barcelona

A quienes me han enseñado que las
matemáticas nunca dejan de aprenderse
y jamás dejarán de sorprendernos.

"Las matemáticas son uno de los descubrimientos de la humanidad. Por lo tanto no pueden ser más complicadas de lo que los hombres son capaces de comprender"

Richard Feynman (Estados Unidos, 1918-1988)

Resumen

El objetivo principal de este trabajo es, por un lado, conocer cómo se organiza el buscador de Google para devolver una clasificación por rangos de todas las páginas web censadas en la actualidad, al introducir una palabra clave en su casilla de búsqueda. Por otro lado, se pretende extraer esta metodología (llamada PageRank) y demostrar que puede aplicarse para realizar una clasificación de cualquier otro tipo o en cualquier otro ámbito.

Para mostrar de manera sencilla y fácilmente entendible dicho funcionamiento, se aplica este método para clasificar cuatro universidades politécnicas que en noviembre de 2015 se unieron para formar una asociación llamada UP4. Estas universidades son la Universitat Politècnica de Catalunya (UPC), la Universidad Politécnica de Madrid (UPM), la Universidad Politécnica de Cartagena (UPCT) y la Universidad Politécnica de Valencia (UPV).

El análisis del funcionamiento del PageRank está basado en grafos y sistemas matriciales, siendo necesarios conocimientos concretos como aquellos relacionados con las matrices positivas y las cadenas de Markov. Esto permite que sea explicado gráficamente y, en el momento de realizar los cálculos pertinentes, ser trasladados al lenguaje de MATLAB para llegar así fácilmente a los resultados esperados.

Finalmente, los resultados obtenidos al final de la aplicación de este método a un caso concreto de clasificación por rangos han sido los esperados: poder realizar un ranking de universidades teniendo en cuenta algunos factores que permiten relacionarlas unas con las otras, tal y como Google actúa con las distintas páginas web. Los cálculos y los pasos previos realizados pueden ser aplicables a otros tipos de clasificación y al número de elementos a clasificar deseado, sea cual sea.

RESUM

L'objectiu principal d'aquest treball és, per una banda, conèixer com s'organitza el cercador de Google per retornar una classificació per rangs de totes les pàgines web censades a l'actualitat, a l'introduir una paraula clau a la seva casella de cerca. Per altra banda, es pretén extreure aquesta metodologia (anomenada PageRank) i demostrar que pot aplicar-se per realitzar una classificació de qualsevol altre tipus o a qualsevol altre àmbit.

Per mostrar de manera senzilla y fàcilment entenedora dit funcionament, s'aplica aquest mètode per a classificar quatre universitats politècniques que al novembre de 2015 van unir-se per formar el que se'n diu la associació UP4. Aquestes universitats són la Universitat Politècnica de Catalunya (UPC), la Universidad Politécnica de Madrid (UPM), la Universidad Politécnica de Cartagena (UPCT) i la Universidad Politécnica de Valencia (UPV).

L'anàlisi del funcionament del PageRank està basat en grafs i sistemes matricials, essent necessaris coneixements concrets com aquells relacionats amb les matrius positives i les cadenes de Markov. Això permet que sigui explicat gràficament i, en el momento de realizar els càlculs pertinents, ésser traslladats al llenguatge de MATLAB per arribar així fàcilment als resultats esperats.

Finalment, els resultats obtinguts al final de l'aplicació d'aquest mètode a un cas concret de classificació per rangs han sigut els esperats: poder realitzar un ranking d'universitats tenint en compte alguns factors que permeten relacionar-les unes amb les altres, tal i com Google actua amb les diferents pàgines web. Els càlculs i els passos previs realitzats poden ésser aplicables a altres tipus de classificació i al nombre d'elements a classificar desitjat, sigui quin sigui.

ABSTRACT

On one hand, the main objective of this paper is to know how the Google search engine is organized to return a classification by range of all the currently censed webs when a key word is used. On the other hand, this paper will try to demonstrate that this methodology (called PageRank) can be applied in other fields.

To explain in an understandable way this procedure, this method is applied to classify four universities which created on November 2015 an association called UP4. These universities are the Universitat Politècnica de Catalunya (UPC), the Universidad Politécnica de Madrid (UPM), the Univerdad Politécnica de Cartagena (UPCT) and the Universidad Politécnica de Valencia (UPV).

The analysis of PageRank's functionality it's based on graphs and matrix systems, where knowledge about positive matrix and Markov chains is necessary. Thanks to this, it can be explained graphically and translated into MATLAB language, in order to get to the expected results.

Finally, the results obtained in the end of this method applied in a range classification, are the expected ones: be able to rank universities by some factors that allow to relate them, just as Google does with different web pages. The calculations and previous steps can be applied in other types of classifications and any number of elements.

Sumario

1. Introducción

En la actualidad podemos encontrar clasificaciones por rangos (o ranking) para todo tipo de sujetos que pueden ser individuos como por ejemplo deportistas, o bien organismos como por ejemplo Universidades, países, empresas, sociedades, asociaciones, etc.. Muchas de estas clasificaciones admiten un tratamiento matemático a partir de propiedades espectrales de una matriz que combina las relaciones entre las entidades. Un ejemplo sería el caso de redes sociales para detectar centros de influencia o la ordenación de páginas web según su importancia.

El diseño de un buscador en la red es un problema de ingeniería matemática. Primeramente se debe traducir el contexto a un modelo matemático y a continuación analizar el modelo con sus conclusiones. Además, tras ese proceso se requiere también un pormenorizado diseño computacional para una eficiente implementación.

Actualmente, el buscador Google es el buscador web más usado y que mayor cantidad de visitas abarca. Éste usa el método de clasificación llamado PageRank, el cual permite, mediante una relación de importancias entre las diferentes páginas censadas, determinar una clasificación usando herramientas matemáticas del Álgebra Lineal.

Llegar a mostrar de una manera simplificada y sencilla cómo actúa un buscador como es Google, que tiene tal cantidad de páginas web a su alcance y que debe tener en cuenta cantidades muy elevadas de factores y criterios que influyen en su estudio y posterior determinación de la clasificación por rangos, sería una gran ventaja a la hora de mostrar una aplicación del Álgebra Lineal con la cual hoy en día todos estamos familiarizados.

Durante toda la etapa educativa se enseñan las matemáticas de un modo más o menos práctico, aprendiendo conceptos y reglas básicas que permiten conocer aspectos cada vez más complejos. Cuanto más se profundiza en el mundo matemático, más abstractos parecen sus razonamientos y, como consecuencia, más complicada parece a simple vista su comprensión o aplicación.

La informática es un campo de la ingeniería que está creciendo en contenidos exponencialmente gracias a los avances tecnológicos y a la investigación, y cada vez más se está usando para resolver problemas cuotidianos o hasta ahora imposibles de solucionar manualmente. Tras esta rama de la ingeniería se esconde una gran cantidad de relaciones, propiedades y teorías matemáticas que hacen posible su funcionamiento.

Es por esto que conseguir demostrar que con conocimientos relativamente sencillos de las matemáticas y, más concretamente, del Álgebra Lineal (por sencillos se entienden los conocimientos de un alumno de primero de carrera y que pueden ser entendibles por alumnos de bachillerato) sea posible entender el funcionamiento de un proceso informático que actualmente es usado por todos, puede hacer resultar tanto motivador cómo sencillo y práctico el estudio de los sistemas matriciales y de las propiedades de las matrices positivas.

1.1. Objetivos del proyecto

A pesar de la complejidad que supone para muchos el mundo de Internet y las infinitas relaciones que hay entre las distintas redes y páginas web, las matemáticas permiten crear un modelo que represente dichas relaciones y entender así el funcionamiento que hay tras, por ejemplo, una búsqueda de Google.

Este proyecto pretende, primeramente, estudiar y analizar el funcionamiento del buscador de Google, es decir, la metodología usada por este buscador para realizar la clasificación por rangos de las distintas páginas al realizar una búsqueda. Determinados conocimientos de Álgebra lineal son de gran utilidad para llegar a comprender tal funcionamiento y la organización interna del buscador.

Una vez entendido y determinado el funcionamiento llevado a cabo por Google, se pretende externalizar el modelo usado para así modificarlo o aprovecharlo para una finalidad distinta, haciéndolo algo más sencillo, y así demostrar que puede ser usado por otros campos y con otros fines. Más concretamente, en este proyecto se determina un modelo que, con la ayuda del programa MATLAB, permite aplicarse a cualquier situación en la que dispongamos de una serie de factores que determinan la importancia de aquello que se quiere clasificar. Así, se usa este modelo para realizar una clasificación de las cuatro universidades politécnicas españolas que se han unido en los últimos meses para crear la unión UP4: las universidades politécnicas de Cataluña (UPC), Valencia (UPV), Cartagena (UPCT) y Madrid (UPM).

1.2. Alcance del proyecto

Debido al limitado número de horas marcadas para la realización de este trabajo (12 créditos ECTS), el alcance de este proyecto es el análisis de la componente matemática que subyace en la clasificación por rangos del buscador Google para aplicarlo a la clasificación de las cuatro universidades que componen UP4. El objetivo es mostrar que este método es aplicable para hacer otro tipo de clasificaciones del mismo tipo, pero manipulando un mayor número de datos. A la vez, mostrar que los conocimientos base que se aprenden en Álgebra lineal en primero de carrera permiten, estudiando más a fondo un tipo concreto de matrices llamadas matrices positivas, llegar a entender el funcionamiento de algunos elementos muy importantes en la vida cotidiana dentro del campo de la informática, como son los buscadores web.

2. El buscador Google

El buscador Google fue diseñado en 1998 por Sergei Brin y Lawrence Page, que en aquel momento eran dos estudiantes de doctorado en Informática de la Universidad de Stanford en la que previamente Brin se había graduado en matemáticas, y Page en Informática.

Figura 1. Sergei Brin y Lawrence Page, creadores de Google (Fuente [10])

El curioso nombre del buscador viene dado por el término *googol*, que fue inventado para referirse al número 10^{100}. Refiriéndonos a escalas gigantescas, cabe destacar que ya cuando se empezó a trabajar con Google había censadas en torno a los 100 millones de páginas web, siendo el número de visitas diarias del buscador entonces más usado de 20 millones. Este valor permite darse cuenta de la grandeza de Google, dado que actualmente atiende 200 millones de consultas diarias e indexa varios miles de millones de páginas web.

Es razonable decir, dadas estas últimas cifras, que es preciso un método eficaz y ordenado que permita al usuario usar con facilidad el buscador y dar con la respuesta que más se adecue a su búsqueda en el menor tiempo posible. Es en este momento en el que empiezan a ser imprescindibles las matemáticas, concretamente el Álgebra lineal, que permitirá analizar, relacionar y ordenar todas las páginas web siguiendo aquellos criterios que se quieran imponer.

2.1. La necesidad del Álgebra lineal en las búsquedas Google

Uno de los factores más sorprendentes de Google es la capacidad, como se ha dicho anteriormente, de ordenar los resultados de las búsquedas de modo que el usuario pueda encontrar lo que desee en, cómo máximo, las diez primeras entradas.

El cómo se organiza Google para resultar tan eficiente reside en el Álgebra lineal. Esta rama de las Matemáticas permite organizar de un modo práctico todas las páginas web y relacionarlas entre ellas, partiendo de dos factores, recopilando toda la información de cada una de ellas: las veces que el elemento buscado aparece en la página, la distancia en que aparece el mismo elemento repetido, si éste aparece en el título o simplemente aparece citado en el texto alguna vez, etc.

Teniendo en cuenta toda esta información, es posible realizar un “grafo dirigido” que relacionará las páginas entre ellas, como explicaremos a continuación.

2.1.1. Organización de las páginas mediante grafos dirigidos

Antes de centrarnos en cómo Google usa los grafos para su estructuración, será necesario describir brevemente el concepto de grafo y entender por qué facilita la manera de ver y entender el funcionamiento del buscador.

Como es de suponer, un grafo es un elemento muy usado en muchas de las aplicaciones del Álgebra lineal, dado que permite representar gráficamente cómo se relacionan diversos elementos, sea en el campo de las Matemáticas o en cualquier otro ambiente nada relacionado con esta ciencia. Es entonces lógico que la definición de éste esté presente en muchos de los libros que hablan o estudian esta rama de la ciencia.

En un lenguaje informal, un grafo consiste en un conjunto de vértices o nodos con algunas conexiones entre ellos, llamando a estas conexiones aristas. Una arista puede conectar dos nodos o un nodo consigo mismo.

Para dar una definición concisa de grafo, hemos escogido una de entre las múltiples publicaciones en abierto que sobre este tema han editado muchas Universidades. En particular, hemos escogido, por su claridad, la definición que se encuentra en uno de los libros de Matemáticas publicado por la Universidad de Salta (Argentina) [3]:

“Un grafo es un triple (N, A, P) donde N es un conjunto de nodos, A es un conjunto de aristas, y P es una función de las aristas tal que cada P(a) = {p,q} donde p,q son nodos (posiblemente con p = q, así que podemos decir que P(a) es un conjunto de 1 o 2 elementos).”

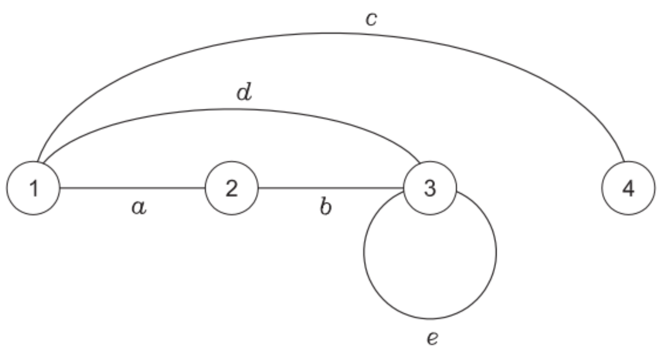

Figura 2 Grafo de nodos 1-4 y arista a-e.(Fuente [12])

En la figura anterior, siguiendo la definición, podríamos distinguir las aristas a,b,c,d y e, que relacionan los nodos 1,2,3 y 4 entre ellos o consigo mismos.

Una vez vista una idea general de grafo, para entender nuestro caso concreto del funcionamiento del buscador, deberemos centrarnos en un tipo especial de grafos llamados "grafos dirigidos". Como bien se entiende en su nombre, es un grafo donde todas sus aristas tienen una dirección, es decir, que relacionan los nodos mediante aristas que salen de un nodo concreto y llegan a otro. Véase la siguiente figura como un ejemplo de grafo dirigido (Figura adaptada para nuestro caso, de grafos presentados en [7]).

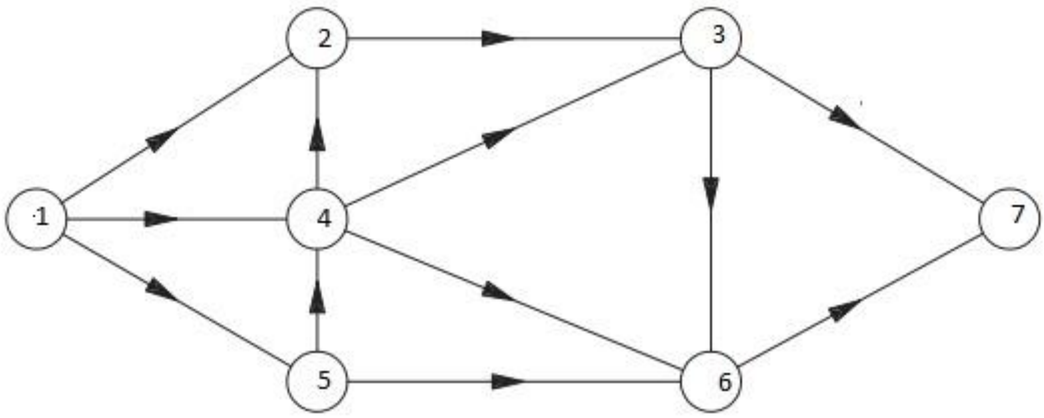

Figura 3 Grafo dirigido.

De un modo parecido, Google relaciona las páginas web, donde cada una de ellas es un nodo P_j, y siendo las aristas los enlaces que llevan de una a otra. Es decir, por ejemplo, que si una arista une el nodo P_i con el nodo P_j, entonces hay un enlace desde la página i que lleva a la página j. Esta esquematización gráfica permite, por lo tanto, poder determinar a partir de qué páginas se puede acceder a otras y así ver el camino que se recorre para llegar a la página que mejor se ajusta a la búsqueda del usuario. Pero si lo que se pretende es dar un orden, un rango de páginas ordenadas según su prioridad, será necesario determinar un factor que nos sirva para escoger el orden adecuado. Siguiendo P. Fernández en [5], llamaremos a este factor la “importancia”.

Esta importancia será, entonces, un valor contenido entre el 0 y el 1 que se dará a la página teniendo en cuenta toda la información de la que se ha hablado en la introducción de este capítulo. Hay dos factores que determinarán que una página se considere de alta importancia:

- Por un lado, serán de “gran importancia” las páginas muy citadas desde otras páginas.

- Por otro lado, se considerarán importantes aquellas páginas poco citadas, pero hecho desde páginas importantes.

Una vez incluido a este análisis el término de la importancia, pasará a complicarse la posibilidad de seguir representando las relaciones entre páginas mediante grafos dirigidos, y será de gran ayuda pasar a otro método de representación, el método matricial.

Para esto, será necesario introducir al lector en el mundo de las matrices para poder entender cómo ayudarán a analizar el procedimiento usado por Google.

2.1.2. Organización de las páginas mediante matrices

El concepto de matriz es clave en el campo del Álgebra lineal y está muy claramente definido en el mundo de las Matemáticas, por lo que en todos los libros de Álgebra lineal se encuentra el término de matriz definido de forma similar. Adaptaremos en este caso la definición que se halla en el libro Herramientas de álgebra lineal y matricial para la ingeniería [6], y que damos a continuación:

“Llamaremos *matriz* de orden *n x m* a coeficientes en un cuerpo conmutativo K a un conjunto de n · m elementos del cuerpo distribuidos en n filas y m columnas y que representaremos

$$A = \begin{pmatrix} a_{11} & \cdots & a_{1m} \\ \cdots & & \cdots \\ a_{n1} & \cdots & a_{nm} \end{pmatrix} \quad , \text{ donde } a_{ij} \in K$$

De este modo, el elemento a_{ij} estará en la fila *i* y en la columna *j*.”

Más adelante, en el apartado 3, se entrará a estudiar más detalladamente el tipo de matrices que se usan para este tipo de organización. Para lo que en este punto necesitamos, que es cómo se entiende la relación entre páginas teniendo en cuenta los dos factores nombrados anteriormente, bastará con entender cómo se representa un grafo a través de una matriz y cómo se distribuyen sus componentes en ella.

De este modo, vamos a definir la matriz M asociada a nuestro grafo que en este caso concreto se necesita:

- Definiremos m_{ij} como un indicador de si la página *j* contiene un enlace que lleva a la página *i*. Es decir, si se puede ir de $j \rightarrow i$. El valor de este indicador m_{ij} será 1 si es posible ir de *j* a *i*, y será 0 en caso contrario.

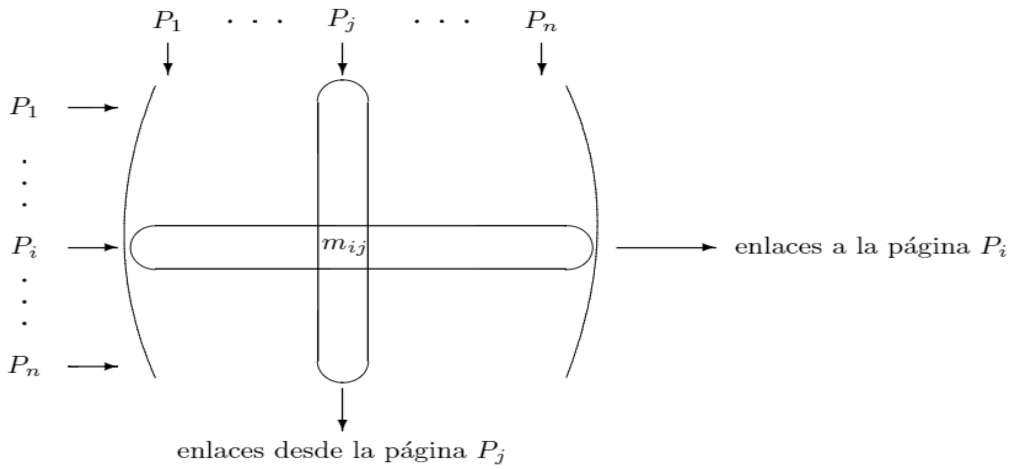

Figura 4 Matriz de relaciones entre páginas

En la figura anterior (Fuente [5]), se puede observar cómo se enlazan las páginas entre sí.

Así por ejemplo, en el grafo dado en la figura 2, la matriz asociada es

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 \end{pmatrix}$$

En la que se puede observar cómo se relacionan los nodos del grafo.

Como ya se ha dicho anteriormente, se quiere determinar la importancia de cada una de las páginas teniendo en cuenta cuántas páginas enlazan a ella y la importancia de las páginas que enlazan a ella. Para ello, tendremos que introducir algún tipo de parámetro que permita cuantificarla.

- Definiremos, por tanto, unos valores x_j que representaran la importancia de cada página P_j que serán, así mismo, proporcionales a la suma de las importancias de las páginas que enlazan con ella.

De este modo, tendremos las relaciones siguientes:

$$x_j = K * \left(\sum x_{\text{páginas que enlazan con } j}\right)$$

donde K es una cierta constante de proporcionalidad.

El sistema de ecuaciones es lineal por lo que podremos representarlo en forma matricial (ver [6]). En este caso concreto, la matriz del sistema es del estilo de la definida en [5], y que se muestra a continuación:

$$\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = K \begin{array}{c} \begin{array}{cccccccccccccccc} P_1 & P_2 & & & & P_{25} & & & & P_{256} & & & & P_{n-1} & & \\ \downarrow & \downarrow & & & & \downarrow & & & & \downarrow & & & & \downarrow & & \end{array} \\ \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 & 0 \\ 1 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & 1 & \cdots & 0 & 1 & 0 & \cdots & 0 & 0 & 0 & \cdots & 1 & 0 \end{pmatrix} \end{array} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

Figura 5. Matriz del sistema de ecuaciones de relación de páginas (pág. 120 de [5])

Como muestra la figura, podemos definir el sistema matricial de este modo:

$$Mx = \lambda x$$

Expresado así, vemos como el vector de importancias x es una solución del sistema definido anteriormente.

Un concepto que todo estudiante de Álgebra lineal conoce y estudia es el de vectores y valores propios (o también denominados autovectores y autovalores). El cálculo de los valores propios y de los vectores propios de una matriz tiene gran importancia no sólo en el estudio del Álgebra lineal sino también en estudios de ingeniería, resolviendo problemas como el cálculo de los momentos de inercia y de los ejes principales de inercia de un sólido rígido, o de las frecuencias propias de oscilación de un sistema oscilante.

Es entonces obvio, para cualquier entendido de las Matemáticas, concluir que nos encontramos en un problema de valores y vectores propios, donde el vector de importancias x será un vector propio de la nxn matriz M. Dado que vamos a entrar un poco más en detalles en estos conceptos, será necesario definir ambos con claridad para que cualquier lector pueda entenderlo.

Como puede encontrarse, por ejemplo, en [6] (página 160):

- Un vector v ϵ E es un **vector propio** de la matriz A si v ≠ 0 y existe λ ϵ K tal que

$$Av = \lambda v$$

- Un escalar λ ϵ K es un **valor propio** de la matriz A si existe v ϵ E, v ≠ 0 tal que

$$Av = \lambda v$$

Diremos entonces que v es un vector de valor propio λ y que λ es un valor propio asociado al vector propio v.

También será necesario explicar cómo se encuentran los valores y vectores propios de una matriz M dada, que siguiendo [6] se determinan de la siguiente manera:

- Los vectores propios de valores propios λ de f son los vectores x ϵ E los componentes $x_1,\ldots,x_n$ de los cuales, en una cierta base u, verifican el sistema

$$(A-\lambda I)X = 0,$$

donde

$$X = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$$

Queremos observar que también se pueden introducir los vectores y valores propios desde un punto de vista geométrico viendo a la matriz A como la matriz de un endomorfismo f de E, en la misma base u (ver [6]).

Haciendo un inciso y sin adentrarnos aun en cómo serán estos valor y vector propios, debe destacarse que el modelo planteado no es determinista, ya que si un buscador en un momento determinado está visitando una página no podemos afirmar qué página estará visitando en un instante de tiempo después, aunque sí con que probabilidad estaría en cada una de las páginas relacionadas. Estamos, por tanto, en un caso de modelo probabilístico.

A su vez, el modelo es dinámico, ya que a la página siguiente que visita el buscador podemos aplicar el mismo razonamiento y así sucesivamente.

De este modo, se puede construir una nueva matriz M’ que tenga en cuenta estas probabilidades, considerando unos nuevos términos de la matriz:

$$m'_{ij} = \frac{m_{ij}}{N_j}$$

donde N_j es el número de enlaces desde la página P_j.

Al realizar este cociente se está dando a cada página que enlaza con la página *j* la misma probabilidad de ser consultada que cualquier otra.

Así mismo, por lo tanto, el sistema a resolver pasará a ser el siguiente:

$$M'x = \lambda x$$

Es necesario remarcar un aspecto característico de todas las posibles *M’* que pueden darse en este caso estudiado, y que podría pasar desapercibido, pero que cogerá un elevado grado de importancia más adelante. Se sabe, como acaba de especificarse, que dicha matriz será una matriz de probabilidades, hecho que hace obvio que todos sus elementos serán $m_{ij}' \geq 0$.

Acaba de descubrirse, pues, un dato del que se podrá sacar una conclusión importante para el estudio que se está tratando de realizar. Dicha conclusión la deduce el Teorema de Frobenius (1908-1912), cuyo enunciado avanzamos y que dice lo siguiente:

“Sea A una matriz (cuadrada) con entradas no negativas, A≥0. Si A es irreducible, entonces

a) existe un valor propio (simple) $\lambda>0$ tal que Av = λv, donde el vector propio es v>0. Además, $\lambda \geq$ abs(µ), para cualquier otro valor propio µ de A.

b) cualquier vector propio v ≥ 0 es un múltiplo de v.

c) si hay *k* valores propios de módulo máximo, entonces son las soluciones de la ecuación $x^k - \lambda^k = 0$."

Véase entonces que si se consigue una matriz no negativa irreducible se podrán determinar resultados de mucha utilidad para el posterior análisis del proceso que usa Google. Es por este motivo que será necesario y de gran ayuda entrar en detalle en el estudio de las matrices positivas y, dentro de éstas, las que son irreducibles, así como de sus propiedades.

3. Matrices positivas

Como se ha visto en el capítulo anterior, Google se basa en las propiedades de las matrices positivas para llevar a cabo su organización y resolver y explicar de forma lo más sencilla posible cómo las páginas se relacionan entre sí y cómo puede llegarse a ordenar el resultado que el usuario desea obtener.

Deberán quedar claros, antes de determinar cómo se usan tales matrices, algunos conceptos relacionados con las mismas que ayudarán y harán posible la clasificación por rangos.

Para entender tales conceptos será necesario recurrir a definiciones ya publicadas en muchos de los libros que se centran en la rama del Álgebra lineal. Concretamente, para este estudio se citarán aquellas que se encuentran en [4] (También pueden verse en [1] y [2]).

Definición 3.1. Una matriz A ϵ M_{nxm} (R) se dice que es *no negativa* (y lo notamos por A ≥ 0) si todos los elementos de A son no negativos. Esto es, la matriz

$$A = \begin{pmatrix} a_{11} & \cdots & a_{1m} \\ \vdots & & \vdots \\ a_{n1} & \cdots & a_{nm} \end{pmatrix}$$

es no negativa cuando $a_{ij} \geq 0$ para todo i ϵ {1,…,n}, j ϵ {1,…,m}.

Denotaremos por $MN_n(R)$ al conjunto de matrices no negativas de orden n.

Del mismo modo, se dice que una matriz A ϵ M_{nxm} (R) es *positiva* (y lo notamos por A > 0) si todos los elementos de A son positivos. Esto es, la matriz

$$A = \begin{pmatrix} a_{11} & \cdots & a_{1m} \\ \vdots & & \vdots \\ a_{n1} & \cdots & a_{nm} \end{pmatrix}$$

se dice que es positiva cuando $a_{ij} > 0$ para todo i ϵ {1,…,n}, j ϵ {1,…,m}.

En el capítulo anterior, al enunciar el teorema de Frobenius hemos mencionado el concepto de matriz irreducible, vamos ahora a detallarlo.

Definición 3.2. Decimos que una matriz A ϵ M_{nxm} (R) (n ≥ 2) es *reducible* cuando existe una matriz P cuyas columnas son una reordenación de las columnas de la matriz identidad (tal matriz la denominamos *matriz permutación*) de modo que

$$PAP^{-1} = \begin{pmatrix} B & 0 \\ C & D \end{pmatrix}$$

donde las matrices B y D son matrices cuadradas. En caso contrario, se dice que la matriz es *irreducible*.

Esta definición **3.2** puede ampliarse con una anotación que aparece en [5] (página 128), la cual permite entender el concepto de *matriz irreducible* de varias maneras:

Nota a definición 3.2: Que una matriz $A \in M_{nxm}$ (R) ($n \geq 2$) sea *irreducible* implica:

1. No existe ninguna permutación (de filas y columnas) que transforma A en una matriz del tipo

$$\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}$$

   donde A_{11} y A_{22} son matrices cuadradas.

2. La matriz $(I + A)^{n+1}$, donde I es la identidad de orden nxn, tiene todas sus entradas positivas.

3. Si A es la matriz de adyacencia de un grafo, entonces el grafo está *fuertemente* conectado. (Diremos que un grafo esté *fuertemente conectado* cuando hay aristas que llevan de un vértice cualquiera *u* a otro *v*).

Una vez introducidas estas definiciones, vamos a presentar los resultados más importantes acerca de estas matrices y que relacionan algunos de los conceptos que se han visto en este capítulo y el anterior, y que son fundamentales para el estudio que estamos realizando.

Teorema 3.1 (Perron). Una matriz positiva A tiene siempre un valor propio λ_1 que es simple, real, positivo e igual al radio espectral $\lambda_1 = \rho(A)$. Los demás valores propios tienen módulo estrictamente menor. Para este valor propio, existe un vector propio con todas las componentes estrictamente positivas.

Observamos que este teorema hace referencia exclusivamente a las matrices positivas, de hecho si las matrices son *no negativas*, el resultado puede fallar encontrando distintos valores propios de mismo módulo e igual radio espectral.

De este modo, Frobenius generalizó más adelante el teorema al caso de las matrices no negativas, dando lugar al siguiente Teorema y que hemos avanzado en el capítulo anterior:

Teorema 3.2 (Frobenius). Una matriz no negativa e irreducible A tiene siempre un valor propio λ_1, que es simple, real, positivo e igual al radio espectral. Para este valor propio, existe un vector propio con todas las componentes estrictamente positivas. Además, si existen otros valores propios $\lambda_2,\ldots,\lambda_h$, de módulo $\rho(A)$, entonces $\lambda_1,\lambda_2,\ldots,\lambda_h$ son las h raíces distintas de la ecuación $\lambda^h - \rho(A)^h = 0$.

Dicho teorema es aplicable a la matriz del sistema de ecuaciones planteado en el capítulo 2:

$$M'x = \lambda x$$

puesto que *M'* es una matriz no negativa.

Estos dos teoremas hacen referencia al radio espectral $\rho(A)$ de la matriz *A* (el valor mayor de entre todos los valores absolutos de los valores propios). De forma general, tenemos el siguiente resultado que hace referencia a cotas para los valores del radio espectral.

Propiedad 3.1: Para todo valor propio λ de una matriz *A* se cumple:

$$|\lambda| \leq \max\{|a_{11}| + \cdots + |a_{1n}|, \ldots, |a_{n1}| + \cdots + |a_{nn}|\}$$

$$|\lambda| \leq \max\{|a_{11}| + \cdots + |a_{n1}|, \ldots, |a_{1n}| + \cdots + |a_{nn}|\}$$

En particular:

$$\rho(A) \leq \max\{|a_{11}| + \cdots + |a_{1n}|, \ldots, |a_{n1}| + \cdots + |a_{nn}|\}$$

$$\rho(A) \leq \max\{|a_{11}| + \cdots + |a_{n1}|, \ldots, |a_{1n}| + \cdots + |a_{nn}|\}$$

En el capítulo anterior, se ha hablado de una matriz de probabilidad M'x = λx donde sus elementos representaban probabilidades siendo éstos $m_{ij}' = m_{ij} / N_j$. De este modo, los elementos m_{ij}' son valores entre 0 y 1 (por lo tanto no negativos), tales que si sumamos los que están en la misma columna se obtiene como resultado la unidad.

Así, una matriz como M' recibe el nombre de matriz estocástica o de Markov, cuya definición precisa la encontramos en la página 89 de [5] y que viene dada de la siguiente manera, y que acompañamos de algunos teoremas que también serán de gran utilidad:

Definición 3.3. Una matriz A = (a_{ij}) ϵ MN_n(R) se dice que es *estocástica* (por columnas) si la suma de los elementos de cada columna de A valen 1. Es decir,

$$a_{ij} \geq 0, \qquad \sum_{j=1}^{n} a_{ij} = 1 \text{ para cada } i = 1, \ldots, n.$$

Ejemplo 5.3.1. La matriz siguiente es estocástica, ya que la suma de todos los elementos de una columna es igual a la unidad.

$$A = \begin{pmatrix} \frac{1}{2} & \frac{1}{3} & \frac{3}{4} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\ 0 & \frac{1}{3} & \frac{1}{2} \end{pmatrix}$$

Un concepto que hará comprender el paso de grafos a matrices cuando los grafos se rigen por probabilidades será el de *Cadena de Markov*, muy importante en las relaciones por probabilidades entre nodos en grafos, y en el que el Álgebra lineal es una fuerte herramienta.

Propiedad 3.2: Sea A una matriz de orden n estocástica, entonces tiene al uno como valor propio. (Página 173 de [7])

Esta propiedad se puede demostrar simplemente viendo que el determinante |A − I| es nulo. Veámoslo:

Consideremos la matriz *A=(a_{ij})* con n=4 y calculemos el siguiente determinante:

$$|A - I| = \begin{vmatrix} a_{11}-1 & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22}-1 & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33}-1 & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44}-1 \end{vmatrix}$$

Si sumamos a la primera fila todas las otras filas, al ser una matriz estocástica, el sumatorio de columnas es la unidad i por tanto, nos dará como resultado una primera fila nula, que anulará a su vez el determinante de ésta. Así, queda demostrada la propiedad.

$$\sum_{j=1}^{4} a_{j1} = \sum_{j=1}^{4} a_{j2} = \sum_{j=1}^{4} a_{j3} = \sum_{j=1}^{4} a_{j4} = 1\,,$$

$$\begin{vmatrix} \sum_{j=1}^{4} a_{j1} - 1 & \sum_{j=1}^{4} a_{j2} - 1 & \sum_{j=1}^{4} a_{j3} - 1 & \sum_{j=1}^{4} a_{j4} - 1 \\ a_{21} & a_{22} - 1 & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} - 1 & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} - 1 \end{vmatrix} = 0$$

3.1. Cadena de Markov

Concerniente a los grafos vistos en el capítulo 2, se ha dicho que cada arista relaciona dos nodos, llevando el nodo *i* al *j* si hay un enlace de la página *i* que lleva a la página *j*. Se ha visto que, cuanto más complejo es el grafo, más merece la pena pasar a representar el problema por medio de sistemas de matrices.

Además, se ha hablado de la transformación del sistema de matrices al añadir probabilidades, haciendo que existan probabilidades que hacen ir desde un nodo a uno o a otro mediante una u otra arista.

En este punto es en el que entra en juego la cadena de Markov, ([4]):

Definición 5.1.1. Una *cadena de Markov* es un proceso aleatorio sin memoria. Esto es, un sistema que evoluciona en el tiempo (k = 0,1,…) está en el estado i_k en el instante k, si en el instante k – 1 está en el instante i_{k-1}:

$$P(X_k = i_k \mid X_0 = i_0, \ldots, X_{k-1} = i_{k-1}) = \\ P(X_k = i_k \mid X_{k-1} = i_{k-1}) = p_{i_{k-1}, i_k}$$

Intuitivamente, esta ecuación implica que, conocido el estado “presente” de un sistema, el estado “futuro” es independiente de su estado “pasado”.

Veamos cómo se describe matricialmente este proceso. Sean $p_1(k), \ldots, p_n(k)$ las probabilidades de los n estados en el tiempo k. Sea $p_{ij}(k)$ la probabilidad de la evolución del estado i al j en el tiempo k, $1 \leq i,j \leq n$. Entonces,

$$\left.\begin{array}{c} p_1(k+1) = p_{11}(k)p_1(k) + \ldots + p_{1n}(k)p_n(k) \\ \vdots \\ p_n(k+1) = p_{n1}(k)p_1(k) + \ldots + p_{nn}(k)p_n(k) \end{array}\right\}$$

lo que puede expresarse matricialmente de la forma:

$$\begin{pmatrix} p_1(k+1) \\ \vdots \\ p_n(k+1) \end{pmatrix} = \begin{pmatrix} p_{11}(k) & \cdots & p_{1n}(k) \\ \vdots & & \vdots \\ p_{n1}(k) & \cdots & p_{nn}(k) \end{pmatrix} \begin{pmatrix} p_1(k) \\ \vdots \\ p_n(k) \end{pmatrix}$$

Este tipo de matrices tienen ciertas peculiaridades o propiedades que serán útiles en este proceso de entender cómo actúa Google en sus búsquedas.

- Como acaba de explicarse, sus columnas suman la unidad, ya que se han dividido las importancias (explicadas en el capítulo anterior) por N_k y hay precisamente N_k valores por fila/columna.

- El vector que se obtiene al resolver el sistema de matricial M'v = λv describirá exactamente con qué probabilidad estará un usuario, tras una unidad de tiempo, en cada una de las páginas de la red habiendo partido de una de ellas P_k.

- Una peculiaridad de esta cadena de *Markov* es que para saber en qué página se encontrará un usuario tras dos instantes de tiempo, sólo será necesario repetir el proceso, es decir, equivalentemente, multiplicar $(M')^2$ por el vector inicial. Análogamente, se podrían determinar los siguientes instantes de tiempo.

Por este último motivo y propiedad, a la matriz *M'* se le da el nombre de *matriz de transición* del sistema. Así, los registros de las sucesivas potencias de la matriz serán las probabilidades de pasar de una página P_j a otra P_i tras varios instantes de tiempo.

Dentro del amplio espacio que agrupa todas las páginas censadas, un estudio realizado por Broder en el año 1999 (véase [3]) concluyó que el 90% de ellas estaba en una gigantesca componente conexa, que a su vez tenía una estructura interna compleja. Por otro lado, afirma que existe un grupo de páginas que no son accesibles desde este gran conjunto y que tampoco llevan a ninguna otra (páginas sin salida).

Éstos casos menos probables pero existentes son los que harían de la matriz M' una matriz con componentes nulas, es decir, dejaría de ser, como se desea, una matriz positiva estrictamente. Así mismo, Google actúa para evitar que esto suceda, y lo hace añadiendo una probabilidad (1-c)·1/n a los vértices, siendo ésta una probabilidad, por decirlo de algún modo, de sorteo, que no viene marcada por la importancia de la página. Cabe remarcar que el parámetro *c* es de un valor comprendido entre 0 y 1, siendo del orden de 0,85 en los cálculos que Google realiza.

De éste modo, la matriz del sistema sería, finalmente:

$$M'' = cM' + (1-c)\begin{pmatrix} p_1 \\ \vdots \\ p_n \end{pmatrix}(1,\dots,1),$$

Observamos que la matriz final se obtiene sumando a la matriz inicial una matriz obtenida como producto de una matriz columna por una matriz fila, dando como resultado en cualquier situación posible, una matriz M'' de probabilidad estrictamente positiva.

Una vez hemos visto que el vector obtenido al resolver este sistema matricial será aquel que determina el estado estacionario de la cadena de Markov que se acaba de construir y, por lo tanto, aquel que determinará en qué página web se encontrará un usuario en un determinado instante, será necesario demostrar y concluir que este vector del que se está hablando es el vector propio *v* dominante.

En nuestro caso particular, puesto que las filas de la matriz suman la unidad, se tiene que 1 es una cota para el radio espectral, y puesto que la matriz es positiva este es valor propio simple y los demás valores propios son en módulo estrictamente menores.

Supongamos los valores propios de la matriz $\lambda_1=1 > |\lambda_2| > \dots > |\lambda_n|$ y que las multiplicidades de λ_i con i distinto de 1, son respectivamente $m_2 \dots m_r$ (claramente $1+m_2 +\dots + m_r =n$).

Consideremos una base de "Jordan", es decir una base en la cual la matriz "por cambio de base" adopta la forma:

$$J = \begin{pmatrix} J_1 & & \\ & \ddots & \\ & & J_r \end{pmatrix}$$

Y cada J_i tiene la forma:

$$J_i = \begin{pmatrix} J_{i_1} & & \\ & \ddots & \\ & & J_{i_r} \end{pmatrix}$$

siendo J_1=(1),

con

$$J_{i_j} = \begin{pmatrix} \lambda_i & & \ddots & 0 & 0 \\ 1 & \lambda_i & \ddots & 0 & 0 \\ \vdots & & \ddots & & \\ 0 & 0 & \dots & 1 & \lambda_i \end{pmatrix}$$

es decir

$$A = SJS^{-1}$$

Ejemplo 5.1.1. La matriz

$$A = \begin{pmatrix} \frac{1}{3} & \frac{1}{5} & \frac{4}{15} \\ \frac{1}{3} & \frac{1}{5} & \frac{4}{15} \\ \frac{1}{3} & \frac{3}{5} & \frac{7}{15} \end{pmatrix}$$

es una matriz estocástica y su forma de Jordan es

$$A = \begin{pmatrix} 1 & & \\ & 0 & \\ & 1 & 0 \end{pmatrix}$$

que se obtiene con la base (v) con v_1 = (4/15, 4/15, 7/15), v_2 = (1,-1, 0),

v_3 =(2/15, 2/15, -4/15).

Entonces, en este caso la matriz A^k se puede calcular de la forma

$$A^k = (SJS^{-1})^k = SJ^k S^{-1}$$

Con

$$J^k = \begin{pmatrix} {J^k}_1 & & \\ & \ddots & \\ & & {J^k}_r \end{pmatrix}$$

y

$$J_i{}^k = \begin{pmatrix} {J^k}_{i_1} & & \\ & \ddots & \\ & & {J^k}_{i_r} \end{pmatrix}$$

siendo

$$J_{i_j}{}^k = \begin{pmatrix} \lambda_i{}^k & & & \\ k\lambda_i{}^{k-1} & \lambda_i{}^k & & \\ \binom{k}{2}\lambda_i{}^{k-2} & k\lambda_i{}^{k-1} & \lambda_i{}^k & \\ \vdots & & \ddots & \ddots \end{pmatrix}$$

Sea ahora un vector $v_0 = c_{11}v_{11}+ c_{21}v_{21} + c_{22}v_{22} + \ldots + c_{31}v_{31}+c_{32}v_{32}+\ldots + c_{r1}v_{r1}+ c_{r2}v_{r2}+\ldots$ que determine las condiciones iniciales del sistema expresado en la base en la que la matriz adopta la forma de Jordan, en cuyo caso si se repite k veces el mecanismo sobre este vector se tiene que el producto resultante es

$$J^k v_0 = c_{11}1^k v_{11} + c_{21}\lambda_2{}^k v_{21} + c_{22}k\lambda_2{}^{k-1}v_{22} + \cdots + c_{31}\lambda_3{}^k v_{31} + c_{32}k\lambda_3{}^{k-1}v_{32} + \cdots + c_{r1}\lambda_r{}^k v_{r1} + c_{r2}k\lambda_r{}^{k-1}v_{r2} + \cdots$$

que claramente, cuando k tiende a infinito y teniendo en cuenta que en módulo los valores propios distintos de la unidad son menores que 1, se tiene que

$$\lim_{k} J^{k} v_0 = c_{11} v_{11}$$

por lo que el vector propio de valor propio 1 determina el estado estacionario del sistema.

Genéricamente las matrices tienen todos los valores propios distintos, en este caso el resultado anterior para el caso genérico queda de la forma siguiente: Consideremos los vectores propios de la matriz $\{v_1, v_2, \ldots, v_n\}$ que determinan, por tanto, una base del espacio. Dado un vector cualquiera del espacio tendrá una expresión en dicha base $v_0 = c_1 v_1 + c_2 v_2 + \ldots + c_n v_n$. Si este vector determina las condiciones iniciales del sistema, se obtendrá:

$$Av_0 = c_1 \lambda_1 v_1 + c_2 \lambda_2 v_2 + \cdots + c_n \lambda_n v_n$$

Como se ha mencionado con anterioridad, al repetir el mecanismo *k* veces podrá obtenerse el sistema que representa el instante *k*:

$$A^k v_0 = c_1 {\lambda_1}^k v_1 + c_2 {\lambda_2}^k v_2 + \cdots + c_n {\lambda_n}^k v_n$$

Si se supone que $c_1 \neq 0$, y habiendo definido λ_1 como el máximo de los valores propios, que se ha demostrado por la propiedad 3.2 que como la matriz A es estocástica este valor propio será de valor 1, se puede realizar el paso siguiente:

$$A^k v_0 = c_1 v_1 + c_2 {\lambda_2}^k v_2 + \cdots + c_n {\lambda_n}^k v_n$$

Cuando se habla de estado estacionario, se dice que el tiempo y por tanto el instante *k* tiende a infinito, por tanto al igual que en el caso general, obtenemos el resultado:

$$\lim_{k} J^{k} v_0 = c_1 v_1$$

He aquí uno de los resultados más importantes del estudio presente: el vector propio v_1 asociado al máximo valor propio λ_1 es aquel que determinará el estado estacionario de la cadena de Markov.

De este modo, se puede concluir que para determinar el estado estacionario dentro de una cadena de Markov, siendo en este caso concreto una búsqueda en Google pero pudiéndose aplicar, como se verá en capítulos posteriores, a muchos otros campos, será necesario resolver el sistema estudiado en este capítulo y encontrar dichos valor y vector propio asociado.

4. Herramienta de cálculo. MATLAB

4.1. Introducción al MATLAB

El nombre de la herramienta MATLAB viene de la abreviatura de 'MATrix LABoratory', es decir, laboratorio de matrices. Esta herramienta de software matemático fue creada por el matemático y programador de computadoras Cleve Moler en 1984, con la idea de poder ser usado en el ámbito del Álgebra lineal y el análisis numérico.

Figura 6. Cleve Moler, creador de la primera versión de MATLAB en 1984. (Fuente [8])

Puede decirse que MATLAB usa un lenguaje propio que es interpretado y puede ejecutarse tanto en el entorno interactivo, como a través de un archivo de script.

MATLAB es una herramienta que facilita muchos cálculos que necesitan realizarse al resolver problemas de campos diversos, entre ellos el que interesa para este trabajo, el matemático o ingenieril. Antes de la creación de este tipo de programas, muchos problemas quedaban sin resolver debido a la dificultad de resolución o a la magnitud y complejidad de los cálculos que debían realizarse. (ver [9] y [11]).

Por ejemplo, centrándonos un poco más en la rama en que se basa este trabajo, el Álgebra lineal, podemos imaginarnos un problema que necesite resolverse mediante sistemas de matrices. Si la dimensión de la matriz o las matrices con las que se desea trabajar es pequeña, puede llevarse a cabo su cálculo manualmente o con herramientas de trabajo sencillas, así como calculadoras u otros aparatos de cálculo. Sin embargo, cuando el cálculo que desea realizarse conlleva trabajar con matrices de grandes dimensiones, es impensable y prácticamente imposible realizarlos. Así, con la creación de herramientas como MATLAB, se ha podido avanzar y ampliar el campo en el que se puede trabajar, haciendo más sencillos y gráficos los métodos de trabajo.

Para la creación de Google, como llegados a este punto puede deducirse, ha sido imprescindible el uso de programas informáticos que permitieran la representación matricial de todos los datos recogidos y almacenados de cada página web para el estudio posterior. Además de la representación de dichos datos, se ha visto que se necesita realizar ciertas operaciones con matrices que las herramientas como MATLAB hacen posible.

Así mismo, para aplicar el método de clasificación por rangos usador por Google (*PageRank*) al caso de estudio de este trabajo, también se hará uso del MATLAB, detallando el procedimiento usado en los siguientes apartados.

4.2. Funciones de MATLAB creadas

Para llevar a cabo la aplicación del procedimiento usado por Google, se han creado tres funciones de MATLAB que permiten mostrar y verificar dicho funcionamiento con algunos ejemplos. Estas funciones, cuya explicación se muestra a continuación, se encuentran especificadas y brevemente comentadas en el anexo (ver página 69).

Función 'vapvep'

Esta función devuelve, dada una matriz estocástica A, el valor propio máximo y su vector propio asociado (ya normalizado). Usa la función ya definida en MATLAB *eig(A)* para determinar la matriz G cuya diagonal son los valores propios y la matriz V cuyas columnas son los diferentes vectores propios asociados a cada valor propio de dicha matriz de entrada.

Además, para que devuelva exactamente lo que se pretende, se escoge el valor propio máximo y se normaliza el vector propio asociado a éste.

Función 'potencias'

Dados una matriz A y un vector v_0 cuyos componentes sumen la unidad, la función potencias determinará el tiempo (referido a cuantas iteraciones serán necesarias para llegar al objetivo marcado) necesario para llegar a obtener el vector límite vep_lim referido al vector propio, si se va ascendiendo el orden de la potencia k de la matriz A progresivamente. Lo que calcula esta función es la diferencia entre el vector que calculamos con el producto A^k*v_0 y el vector propio del valor propio máximo al que se ha demostrado que se debe llegar. Cuando esta diferencia es suficientemente pequeña, se dice que se ha logrado alcanzar el objetivo, y devuelve el número de iteraciones que han sido necesarias.

Función 'clasif_uni'

La función '*clasif_uni'* es aquella que recibe como entradas una matriz *M* y tres escalares *u*, *f* y *d*, siendo la matriz *M* la que contiene las relaciones entre diferentes universidades teniendo en cuenta diferentes factores (matriz que se deduce en la explicación de la página 54), *u* y *f* dos escalares que se refieren al número de universidades que se quiere clasificar y la cantidad de factores que se tienen en cuenta, respectivamente; y finalmente siendo *d* el escalar que determina la proporción de información que se considera que da dicho estudio en relación a la cantidad total de factores que podrían tenerse en cuenta, es decir, el parámetro *d* mencionado en la página 55.

Así, esta función realizará los cálculos necesarios (expuestos y explicados en el anexo, en la página 71) para acabar determinando, haciendo uso de la anterior función 'vapvep' detallada, un vector '*clasif'* en el que cada elemento representa, a su vez, un valor asociado a cada una de las universidades estudiadas. De este modo, la universidad cuyo valor asociado sea máximo, será la universidad que ocupará la primera posición en el ranking, y así sucesivamente.

5. Aplicación del MATLAB

Para introducir el caso práctico que se estudiará y realizará en este proyecto, se efectuará una muestra o ejemplo para confirmar que, efectivamente, el procedimiento usado por Google para realizar sus búsquedas puede ser usado en escenarios muy distintos y campos muy diversos.

Concretamente, ese ejemplo consistirá en estudiar el comportamiento de una rata que se mueve entre los compartimentos de un laberinto. Veamos dos casos con dos ejemplos de laberintos distintos:

5.1.1. Ejemplo 1: "Laberinto para ratas"

- ***Laberinto 1º:*** En este primer caso, la rata se mueve por el laberinto de la figura 5 que tiene los 4 compartimentos 1, 2, 3 y 4. Supongamos que en determinados instantes de tiempo t = 0,1,2,… la rata elige o bien una de las salidas del compartimento en el que se encuentra, o bien permanecer en él, todo ello con la misma probabilidad. De este modo, la rata tendrá en cada instante ciertas opciones equiprobables.

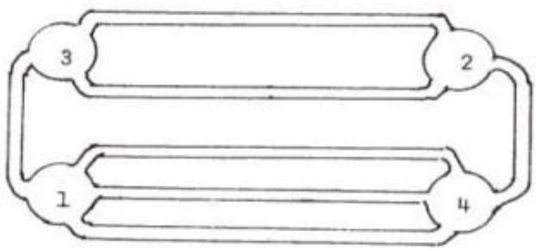

Figura 7. Laberinto 1º(Fuente [18])

Teniendo en cuenta todo lo expuesto a lo largo de los capítulos anteriores, podemos afirmar que estamos en un claro caso de cadena de Markov y, por lo tanto, podremos determinar mediante el procedimiento determinado y usado por Google el estado estacionario del sistema o problema estudiado.

Construyamos, pues, la matriz estocástica del caso actual:

$$P = \begin{pmatrix} \frac{1}{5} & 0 & \frac{1}{4} & \frac{3}{5} \\ 0 & \frac{1}{4} & \frac{1}{2} & \frac{1}{5} \\ \frac{1}{5} & \frac{1}{2} & \frac{1}{4} & 0 \\ \frac{3}{5} & \frac{1}{4} & 0 & \frac{1}{5} \end{pmatrix}$$

Realizando el cálculo con Matlab y con la ayuda de la función 'vapvep.m' descrita anteriormente, se obtiene el valor propio máximo siguiente y su vector propio asociado (normalizado):

```
P =

    0.2000         0    0.2500    0.6000
         0    0.2500    0.5000    0.2000
    0.2000    0.5000    0.2500         0
    0.6000    0.2500         0    0.2000

>> [vap,vep]=vapvep(P)

vap =

    1.0000


vep =

    0.2778
    0.2222
    0.2222
    0.2778
```

Observamos cómo, al ser una matriz estocástica, el radio espectral obtenido (valor propio de valor absoluto máximo) es la unidad, tal y como ya se esperaba después de lo visto en las explicaciones del capítulo anterior. Así, el vector asociado a este valor propio es el que nos determinará el estado estacionario del sistema.

- ***Laberinto 2º:*** Este segundo ejemplo tratará de solucionar o resolver una situación similar anterior, pero en este caso con un laberinto distinto, siendo el actual laberinto el mostrado en la figura 6. Ahora, como se observa en dicha figura, el laberinto consta de tres compartimentos A, B y C.

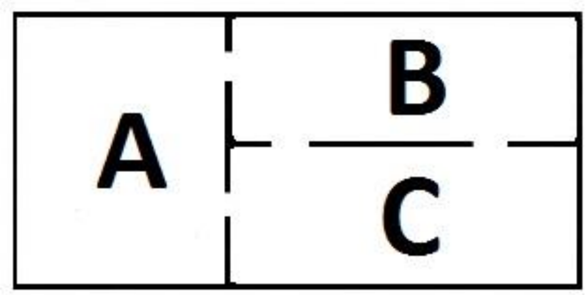

Figura 8. Laberinto 2º

Cabe destacar en este caso, que si se lanza la rata al aire, no tendrá la misma probabilidad de caer en una cámara del laberinto que en otra debido a que las superficies no son las mismas. Una posible distribución de probabilidades podría ser, entonces, inicialmente, p = (0,37, 0,30, 0,33), porque en estas proporciones están aproximadamente las áreas de las cámaras.

Para este caso, la matriz de transición de la cadena de Markov sería la siguiente:

$$P = \begin{pmatrix} 0 & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{2} & 0 & \frac{2}{3} \\ \frac{1}{2} & \frac{2}{3} & 0 \end{pmatrix}$$

Procediendo a aplicar el mismo método que el caso del Laberinto anterior, obtenemos en este caso los siguientes resultados (usando también Matlab):

```
P =

         0    0.3333    0.3333
    0.5000         0    0.6667
    0.5000    0.6667         0

>> [vap,vep]=vapvep(P)

vap =

    1.0000


vep =

    0.2500
    0.3750
    0.3750
```

5.1.2. Ejemplo 2: Potencias de una matriz

Para ejemplificar la función de la matriz *potencias* partiremos del vector v_0 y la matriz A estocástica siguientes:

$$v0 = \begin{pmatrix} 0.2 \\ 0.3 \\ 0.4 \\ 0.1 \end{pmatrix}$$

$$A = \begin{pmatrix} 0.4 & 0.1 & 0.1 & 0.2 \\ 0.4 & 0.6 & 0.3 & 0.1 \\ 0.15 & 0.2 & 0.4 & 0 \\ 0.05 & 0.1 & 0.2 & 0.7 \end{pmatrix}$$

Veamos cómo, siguiendo los pasos comentados anteriormente al explicar esta función en MATLAB, se determina el número necesario de iteraciones (de potencias de la matriz A) para llegar a un vector suficientemente cercano al vector propio asociado al valor propio máximo.

```
A =

    0.4000    0.1000    0.1000    0.2000
    0.4000    0.6000    0.3000    0.1000
    0.1500    0.2000    0.4000         0
    0.0500    0.1000    0.2000    0.7000

>> v0= [0.2;0.2;0.4;0.2]

v0 =

    0.2000
    0.2000
    0.4000
    0.2000

>> [d,n] = potencias(A,v0)

d =

   4.9591e-05


n =

     7
```

Véase cómo, efectivamente, se acaba llegando (después de 7 iteraciones en este caso) a un vector cuya diferencia respecto al vector propio al que se pretendía llegar es suficientemente pequeña para considerarlo igual.

6. Trabajo de aplicación

Una vez estudiado el procedimiento que Google usa para determinar un cierto ranking dentro de todo el repertorio de páginas webs existentes en función, como ya se ha visto, de ciertos parámetros que dan importancias distintas a cada una de ellas, será interesante ver cómo puede extrapolarse este procedimiento y ser usado en otros ámbitos muy distintos; ya sea en biología como han mostrado los ejemplos anteriores, o bien un ámbito más ingenieril y que se vea implicado más directamente en el entorno más actual de la ingeniería. Concretamente, se pretenderá aprovechar y usar dicho procedimiento para analizar y comparar distintas universidades.

Se podría analizar una gran cantidad de aspectos en el entorno de las universidades, ya sea estudiando la calidad de la docencia, el nivel del alumnado, la antigüedad de las escuelas, etc. Actualmente, existe un elevado número de universidades en todo el mundo y cada una de ellas ofrece un listado amplísimo de estudios diferentes. Dada esta gran adversidad, será de gran utilidad centrarse en un pequeño conjunto de universidades a comparar y en un aspecto en concreto a evaluar y comparar, de modo que sea más sencillo para el lector entender el modo de aplicación de clasificación por rangos estudiado.

Con este objetivo, se ha decidido estudiar un caso concreto que implica a cuatro conocidas universidades españolas. Estamos hablando de la reciente creación de la asociación UP4 que se explicará a continuación, que incluye a las universidades politécnicas de Madrid (UPM), Cataluña (UPC), Valencia (UPV) y Cartagena (UPCT) (Ver [15]).

6.1.1. Asociación UP4. Análisis y comparación de universidades

El pasado 16 de noviembre de 2015, las Universidades Politécnicas de Madrid, Catalunya, Cartagena y Valencia, crearon la asociación UP4, nombre que representa esta unión de cuatro universidades politécnicas.

El objetivo de esta asociación es impulsar la enseñanza e investigación tecnológica de calidad, sumando la excelencia de cada una de ellas para multiplicar sus capacidades y afianzar la colaboración en docencia, investigación y transferencia de conocimiento e innovación.

Esta unión es una clara oportunidad para llevar a cabo un estudio como el que se pretende en este caso, ya que va a permitir, a pesar de haber realizado este acuerdo de unión, valorar y colocar a cada una en un nivel u otro de un ranking, es decir, comparar su importancia y su estatus en función de los parámetros que se decida tener en cuenta.

Una vez analizada la información de la que se dispone referente a las publicaciones, tesis, documentos publicados por todos los departamentos de cada universidad y a los premios otorgados también a cada uno de ellos, entre otros, se han seleccionado los considerados de mayor reconocimiento o importancia. Así, para la realización y estudio de la clasificación por rangos de dichas universidades, se tendrán en cuenta los cuatro factores siguientes:

- *Artículo de revista y capítulo de libro.*
- *Patente de invención.*
- *Premio o reconocimiento.*
- *Tesis doctoral.*

Los diferentes datos recogidos a través de las páginas web de cada una de estas universidades (ver [13], [14], [16] y [17]) se muestran en la siguiente tabla:

	T_1	T_2	T_3	T_4
	Artículo de revista y capítulo de libro	Patente de invención	Premio o reconocimiento	Tesis doctoral
UPC	34356	1121	2737	5524
UPV	20240	271	488	1104
UPM	29867	1035	754	3207
UPCT	2602	12	23	290

Sería un error considerar esta tabla como buena o suficiente para realizar el estudio, ya que es un hecho que cada universidad tiene un número de investigadores distinto, así que hay que tener en cuenta este dato para que los valores que representan los artículos, las patentes, los premios y las tesis de cada universidad puedan compararse unos con otros.

De este modo, se ha rehecho la tabla teniendo en cuenta el número de investigadores de cada universidad y siendo así cada valor de la tabla el número de publicaciones en relación con el número de investigadores. (núm. Publicaciones / núm. Investigadores):

	T_1	T_2	T_3	T_4
	Artículo de revista y capítulo de libro	Patente de invención	Premio o reconocimiento	Tesis doctoral
UPC	10,57	0,35	0,84	1,70
UPV	14,62	0,20	0,35	0,80
UPM	9,81	0,34	0,25	1,05
UPCT	4,10	0,02	0,04	0,46

Como se muestra en la tabla, cada T_i es un factor o criterio de clasificación. Cada factor T_i tendrá un valor distinto para cada universidad. De este modo, podrá determinarse una matriz de probabilidades. Para crear esta matriz de probabilidades, antes se deberá aclarar y tener en cuenta una serie de matrices que se explicarán a continuación.

6.1.2. Procedimiento práctico

Para poder crear esta matriz de probabilidades, igual que en el procedimiento usado por Google, será necesario determinar la relación que tienen entre ellos los diferentes factores. Esta relación vendrá representada por una matriz de ceros y unos, compuesta por diferentes submatrices, donde cada submatriz representará la relación entre los valores que toma cada factor para cada una de las universidades. Es decir, el elemento de la fila *i* y de la columna *j* de la submatriz 1 representará la relación (si es mayor o igual será 1 y si es menor será un 0) que tiene el factor 1 de la universidad *i* con el mismo factor de la universidad *j*. Y así sucesivamente para cada factor analizado. Pongamos un ejemplo para que sea sencillo de entender:

Ejemplo 6.1: Supongamos que los valores de un factor para las 4 universidades son los siguientes [3400, 300, 2678, 34].
La submatriz de este factor sería la siguiente:

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Así mismo se haría para los otros tres factores, de manera que la matriz M finalmente obtenida para el caso que se está estudiando en este trabajo, sin tener en cuenta la proporción de publicaciones con respecto al número de investigadores, será la que se muestra a continuación. En este caso, hemos representado con un asterisco * las relaciones positivas o 1 del elemento m_{ij}, es decir, aquellas en que el factor analizado de la universidad *i* es mayor que el mismo factor en la universidad *j*:

$$M = \begin{pmatrix}
\ast & \ast & \ast & \ast & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \ast & 0 & \ast & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \ast & \ast & \ast & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & \ast & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \ast & \ast & \ast & \ast & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \ast & 0 & \ast & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \ast & \ast & \ast & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \ast & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \ast & \ast & \ast & \ast & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \ast & 0 & \ast & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \ast & \ast & \ast & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \ast & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \ast & \ast & \ast & \ast \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \ast & 0 & \ast \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \ast & \ast & \ast \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \ast
\end{pmatrix}$$

Cabe decir que se podría haber dado diferentes importancias a cada uno de los factores que se estamos analizando. Sin embargo, para comprobar el buen funcionamiento no se ha considerado que fuera necesario.

Vistos los cambios obtenidos en la segunda tabla del subcapítulo anterior con respecto a la relación de dimensión de los valores que representan un mismo factor para cada universidad, será necesario cambiar la matriz M dada anteriormente, ya que presentará algunos cambios. Estos cambios se deberán, como ya se ha mencionado en dicho subcapítulo, a que no todas las universidades tienen las mismas dimensiones y, por lo tanto, el mismo número de investigadores.

$$M = \begin{pmatrix}
* & 0 & * & * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
* & * & * & * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & * & * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & * & * & 0 & * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & * & 0 & * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & * & * & * & * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & * & * & * & * & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & * & * & * & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & * & * & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & * & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & * & * & * & * \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & * & 0 & * \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & * & * & * \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & *
\end{pmatrix}$$

Como puede observarse, la matriz anterior, que tiene en cuenta el número de investigadores de cada universidad, varía en algunos términos con respecto a la matriz M que no tiene en cuenta dicho factor. Así, para realizar el estudio y el ranking de estas cuatro universidades, se cogerá la matriz M que sí que tiene en cuenta el número de investigadores de cada universidad, ya que tratará unas proporciones de los datos más razonables que la primera matriz M descrita. Cabe decir que estas matrices pueden crearse mediante la adición de submatrices en programas como MATLAB.

Cuando se habla de los distintos factores a tener en cuenta, hay un hecho que no hay que pasar por alto, y es que para la creación de esta matriz M sólo se han considerado cuatro factores, cuando en realidad hay otros factores, aunque se consideren de menor peso o importancia y influencia en el estudio actual, que también afectan en cierta medida. Es por este motivo que se considerará un término que denominaremos *d* que permitirá tener en cuenta este porcentaje de factores que no se tienen en cuenta. Así, el término *d* será un valor razonablemente alto aunque sin llegar a la unidad, siendo lógico que los factores que no se tienen en cuenta en el estudio son menos importantes o relevantes.

Más adelante se comprobará la validez o variación de los resultados para distintos valores de *d*, pero se empezará cogiendo y mostrando el procedimiento con una *d* = 0,85, ya que es el valor usado por Google.

De este modo, para obtener una matriz P de probabilidades que represente el estudio a resolver, será necesario realizar el cálculo matricial mostrado a continuación:

$$P = (1-d)*\left(\frac{1}{n}\right) + d*M$$

donde $\left(\frac{1}{n}\right)$ es una matriz estocástica 16x16 que, multiplicada al valor (1-*d*), representa la matriz de los factores que no se tienen en cuenta en el estudio. Esta matriz, en lenguaje MATLAB, se crearía de la manera siguiente, siendo *n* la dimensión de la matriz M:

$$(1) = \ ones(n) * \frac{1}{n}$$

Por otro lado, véase el producto *d**M que hace referencia a los 4 factores T_i de cada universidad.

Esta matriz M, para poder aplicar la teoría expuesta y el método usado por Google, deberá ser también estocástica para que el producto final que da lugar al a matriz P también lo sea. Así, esta matriz deberá sufrir una pequeña variación, así como se hacía en las matrices de Google para conseguir que fueran de probabilidad y estocásticas, es decir, que el resultado de la suma de los elementos de una misma columna fuera la unidad.

Así mismo, la matriz M ya no será una matriz de ceros y unos, sino que deberán dividirse los elementos de cada columna por el valor de la suma de todos los elementos de esta. Es decir, que la matriz M resultará al final la matriz siguiente:

$$M = \begin{pmatrix}
\frac{1}{2} & 0 & \frac{1}{3} & \frac{1}{4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{1}{2} & 1 & \frac{1}{3} & \frac{1}{4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{1}{3} & \frac{1}{4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{1}{4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & \frac{1}{3} & \frac{1}{2} & \frac{1}{4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \frac{1}{3} & 0 & \frac{1}{4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \frac{1}{3} & \frac{1}{2} & \frac{1}{4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{3} & \frac{1}{4} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{4} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \frac{1}{3} & \frac{1}{2} & \frac{1}{4} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{3} & 0 & \frac{1}{4} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{3} & \frac{1}{2} & \frac{1}{4} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{4}
\end{pmatrix}$$

Puede comprobarse como, ahora sí, la matriz M finalmente es una matriz estocástica, dado que la suma de todas sus filas, para cada columna, da la unidad.

Hay que tener presente que dichas matrices las calculará MATLAB, así como la matriz P resultante. Una vez obtenida dicha matriz, será necesario, tal y como muestra el procedimiento usado por la función 'clasif_uni', calcular el vector propio asociado al valor propio máximo 1. A partir de este instante, es importante entender cómo obtendremos la clasificación de las universidades estudiadas, que son cuatro en este caso.

El vector propio asociado al valor propio 1 (valor propio máximo) de nuestra matriz P, será un vector de dimensión nx1, siendo n = 16 en nuestro caso. Así, este vector propio será el que determinara qué resultado o puntuación final obtendrá cada universidad para cada factor asociado. Es decir, que el vector será a su vez composición de tantos vectores como factores se consideren (tantos vectores como submatrices tuviésemos). Así, si se suman los elementos de la posición *k* de cada subvector, se obtendrá la puntuación final asociada a la universidad *k* teniendo en cuenta todos los factores considerados.

De este modo, para el caso de UP4 con los cuatro factores estudiados y un valor d = 0.85, se obtienen finalmente el vector mostrado a continuación. Veamos cómo lo mostraría MATLAB en pantalla:

```
M =

     1     0     1     1     0     0     0     0     0     0     0     0     0     0     0     0
     1     1     1     1     0     0     0     0     0     0     0     0     0     0     0     0
     0     0     1     1     0     0     0     0     0     0     0     0     0     0     0     0
     0     0     0     1     0     0     0     0     0     0     0     0     0     0     0     0
     0     0     0     0     1     1     1     1     0     0     0     0     0     0     0     0
     0     0     0     0     0     1     0     1     0     0     0     0     0     0     0     0
     0     0     0     0     0     1     1     1     0     0     0     0     0     0     0     0
     0     0     0     0     0     0     0     1     0     0     0     0     0     0     0     0
     0     0     0     0     0     0     0     0     1     1     1     1     0     0     0     0
     0     0     0     0     0     0     0     0     0     1     1     1     0     0     0     0
     0     0     0     0     0     0     0     0     0     0     1     1     0     0     0     0
     0     0     0     0     0     0     0     0     0     0     0     1     0     0     0     0
     0     0     0     0     0     0     0     0     0     0     0     0     1     1     1     1
     0     0     0     0     0     0     0     0     0     0     0     0     0     1     0     1
     0     0     0     0     0     0     0     0     0     0     0     0     0     1     1     1
     0     0     0     0     0     0     0     0     0     0     0     0     0     0     0     1

>> clasif = clasif_uni(M,4,4,0.85)

clasif =

    0.6067
    0.2547
    0.0910
    0.0476
```

Habiéndose considerado el orden de universidades de la manera siguiente: [UPC, UPV, UPM, UPCT], el resultado obtenido ordenaría las universidades dentro de un ranking tal y como se muestra a continuación:

1. UPC
2. UPV
3. UPM
4. UPCT

Como se ha comentado anteriormente, en este estudio se ha tenido en cuenta un valor de *d* = 0.85, pero se podría haber cogido otro valor cercano a la unidad que representara esta importancia de los factores escogidos con respecto a los que no se han tenido en cuenta. Será importante comprobar que los resultados no varían si se cogen diferentes valores de *d*. Dicha comprobación se hará mostrando en la tabla siguiente los resultados obtenidos para distintos valores de *d*:

Valor *d* escogido	Vector de puntuaciones obtenido con MATLAB
0.7	0.5017 0.2569 0.1505 0.0909
0.75	0.5333 0.2564 0.1333 0.0769
0.8	0.5682 0.2557 0.1136 0.0625
0.85	0.6067 0.2547 0.0910 0.0476
0.9	0.6494 0.2535 0.0649 0.0323

Vemos como, para distintos valores de *d*, aunque se obtienen puntuaciones de valores distintos, la clasificación que se obtiene de las 4 universidades es la misma para todos los valores probados, colocando a la Universitat Politècnica de Catalunya en primer lugar, seguida de la de Valencia, a continuación de la de Madrid y colocando en última posición a la Universidad Politécnica de Cartagena.

7. Planificación temporal y de costes

7.1. Planificación temporal

Para la realización de este trabajo, ha sido necesaria una planificación específica, ya que han sido unos hechos y aprendizajes concretos los que permitido ir avanzando adecuadamente e ir cumpliendo aquellos objetivos propuestos al inicio del planteamiento del mismo.

Así, el trabajo ha sido planificado como se muestra a continuación:

- Recogida de información sobre Google: Semanas 1-3
- Estudio del Álgebra lineal y de matrices positivas: Semanas 4-7
- Elección de la aplicación del método estudiado: Semanas 6-7
- Búsqueda de datos para la aplicación: Semanas 8-9
- Organización de la aplicación: Semanas 9-11
- Aplicación, MATLAB y análisis de resultados: Semanas 12-17

7.2. Planificación de costes

El coste que ha tenido la realización del trabajo, dado que ha podido realizarse únicamente con el uso de ordenadores, es el coste del software (licencia de MATLAB, paquete Microsoft Office,..) y el del material (ordenador portátil, los cd usados para la entrega del proyecto, fotocopias y impresiones de documentos para seleccionar y buscar información, fotocopias del trabajo realizadas para la entrega del mismo) que han sido necesarios para la realización del mismo, así como el coste de las horas de trabajo. Este último coste resulta del número de horas que suponen los créditos del TFG por el precio de la hora.

Teniendo en cuenta, por lo tanto, todo aquello que ha sido necesario para llevar a cabo este trabajo, el cálculo del coste total se puede ver desglosado en la tabla dada a continuación:

Elementos	Coste
Ordenador HP	700 €
Licencia MATLAB estudiantes	35,00 €
Paquete Microsoft Office 2007	80 €
Fotocopias + 2 x Cd	20 €
12 Créditos TFG	474,36 €
COSTE TOTAL	1309,36 €

Cabe comentar que, como puede observarse, el coste que supone la realización de este trabajo no es muy elevado. Además, hay que tener en cuenta que, hoy en día, puede disponerse de ordenadores que lleven incorporados el paquete Microsoft Office fácilmente en sitios públicos y que, con respecto a la licencia de MATLAB, se ofrecen facilidades para estudiantes, haciendo su uso y instalación más económica.

Conclusiones

Para poder detallar cuáles han sido las conclusiones sacadas de los resultados de este trabajo, es necesario recordar cuál era uno de sus objetivos: mostrar de manera sencilla que los conocimientos adquiridos en el curso de Álgebra lineal permiten llevar a profundizar en algunos de ellos, en este caso las matrices positivas y la cadena de Markov, y así resolver problemas o realizar funciones que actualmente son de gran importancia y son usados por gran parte de la sociedad, como puede ser el buscador Google.

Habiendo analizado el funcionamiento del PageRank y aplicado tanto sus ideas como la manera de relacionar los elementos a clasificar, se ha podido exportar este método a la clasificación de las universidades que forman UP4, teniendo en cuenta algunos factores.

Es precisamente llegado este punto cuando se puede concluir que, usando las funciones y la organización de los distintos elementos y factores que se ha expuesto en este trabajo, podría realizarse una clasificación del mismo tipo siendo el valor de elementos o factores tan elevado como se quisiera. Además, también puede usarse para cualquier otro tipo de clasificación por rangos, como se ha mostrado en algunos ejemplos del trabajo.

Por otro lado, y finalmente, también cabe mencionar la aplicación de este trabajo a la docencia. Una consecuencia de haber realizado un ejemplo de aplicación del Álgebra lineal que simplifica y muestra el funcionamiento de Google es un arma de motivación importante de cara a impulsar a los alumnos a seguir profundizando en dicha materia. Saber que aquello que se estudia tiene una importancia tan grande en algo que hoy en día todos conocemos y usamos a diario es una manera potente de aumentar las ganas de aprender y descubrir más de aquello de lo que ya se tiene una base.

Agradecimientos

En la realización de este trabajo, ha sido necesario un cierto seguimiento y apoyo, o más bien dicho, orientación. A pesar de los conceptos adquiridos en primer curso de carrera en la asignatura de Álgebra lineal, hay ciertos conocimientos que he tenido que aprender, así como seleccionar qué es aquello que iba a dar los frutos esperados en un principio o cómo afectaría a los resultados el modo de realizar o organizar las ideas.

Al realizar un trabajo de investigación y posterior aplicación de aquello aprendido, son de gran ayuda todas aquellas personas que dominan y conocen más a fondo la materia y que, sin dar la solución sin más, ayudan a llegar a ella y a aprender en el esfuerzo. Por todo esto debo agradecer a la tutora del trabajo María Isabel García su seguimiento, paciencia y apoyo a lo largo de la realización de éste, así como por guiarme y enseñarme que aquello que aprendemos de base en los primeros cursos de universidad sirven para mucho más y se encuentra escondido tras muchos problemas cotidianos.

Bibliografía

Referencias bibliográficas

[1] R.B. BAPAT and T.E.S. RAGHAVAN, Nonnegative Matrices and Applications, Encyclopedia of Mathematics and its Applications (No. 64), Cambridge University Press, 1997.

[2] A. BERMAN, R.J. PLEMMONS, Non-negative Matrices in Mathematical Sciences, SIAM. Filadel_a (1994).

[3] BRODER, A. etal.:Graph structure in the web. Accesible en la página web: [www9.org/w9cdrom/160/160.html, 29 de septiembre de 2015]

[4] DOMÍNGUEZ GARCÍA, J.L., GARCÍA PLANAS, M.I. Introducción a la teoría de matrices positivas. Aplicaciones. Iniciativa Digital Politècnica, 2013.

[5] FERNANDEZ, P. *El secreto de Google y el Álgebra lineal.* Madrid: Universidad Autónoma de Madrid, Bol. Soc. Esp. Mat. Apl. nº30(2004), p. 115-141.

[6] GARCÍA PLANAS, M.I., MAGRET PLANAS, M.D. Eines d'àlgebra lineal i matricial per a l'enginyeria. Editado por las autoras. Barcelona, 2014.

[7] [http://matema.ujaen.es/jnavas/web_modelos/pdf_mmb08_09/modelos%20matriciales.pdf, 7 de octubre de 2015]

[8] [http://es.mathworks.com/company/aboutus/founders/clevemoler.html?refresh=true, 2 de enero de 2016]

[9] [http://orion.ciencias.uniovi.es/~riera/modelado/practicas/pract_01.pdf, 12 de noviembre de 2015]

[10] [http://www.edenarius.com/varios/larry-page-y-sergey-brin-fundadores-de-google/, 31 de desembre de 2015]

[11] [http://www.sc.ehu.es/sbweb/energias-renovables/MATLAB/numerico/propios/propios.html, 10 de noviembre de 2015]

[12] [http://www.unsa.edu.ar/~hibbard/discreta/grafos.pdf, 29 de septiembre de 2015]

ETSEIB

[13] [http://www.upc.edu/investigacion/departamentos-e-institutos/departamentos, 18 de diciembre de 2015]

[14] [http://www.upct.es/saladeprensa/cifras_web.pdf, 18 de diciembre de 2015]

[15] [http://www.upm.es/institucional/UPM/CanalUPM/NoticiasPortada/Contenido/95ef8ac433 011510VgnVCM10000009c7648aRCRD, 24 de noviembre de 2015]

[16] [http://www.upm.es/observatorio/vi/index.jsp, 18 de diciembre de 2015]

[17] [http://www.upv.es/entidades/VIIT/menu_urlc.html?/entidades/VIIT/info/U0687647.pdf, 18 de diciembre de 2015]

[18] [http://www.vc.ehu.es/campus/centros/farmacia/deptos-f/depme/profesor/gracia/defi.pdf, 29 de septiembre de 2015]

Bibliografía complementaria

- LAZOVA, V., BASNARKOV, L. PageRank approach to ranking national football teams. Soporte digital, 2015.
- [http://www.shanghairanking.com/es/ARWU-Methodology-2014.html]
- [http://www.unavarra.es/personal/victor_dominguez/libroMatlab.htm]

Anexos

Funciones creadas en Matlab

- Funció vapvep

```
function[vap,vep]=vapvep(A)
    n = length(A); %Determina la dimension de la matriz de entrada
    [V,G] = eig(A); %Asigna a V y G las con los veps y vaps de la
matriz, respectivamente.
    vapmax = max(max(G(:,:))); %Busca el máximo de las columnas de
G,que será la columna con el valor propio máximo.
    vap = abs(vapmax);
    [f,c] = find(G==vapmax); %Encuentra posición que ocupa el valor
propio máximo en la matriz G.
    vepmax = V(:,c); %Coge como vector propio el vector de la columna
de V correspondiente a la columna que ocupa el valor propio máximo en
G.
    vep = (vepmax/sum(vepmax)); %Normaliza el vector seleccionado,
que sera el vep de vap maximo.
end
```

➢ Funció potencias

```
function [d,n] = potencias(A,v0)
    k=1; %Número de potencias a la que queremos elevar la matriz A
    n = length(A);
    E_r = 1*10^-4; %Error que se considera suficientemente pequeño
para aceptar dos vectores iguales.
    [V,G] = vapvep(A);
    vep_lim = G;
    d = 1;
    t = 0;
    while d >= E_r
        vect = (A^k)*v0;
        dist = abs(vep_lim-vect);
        d = max(dist);
        k = k+1;
        t = t+1;
    end
    n = k-1; %Devuelve el número de iteraciones que han sido
necesarias para llegar a conseguir la distancia deseada.
end
```

- Funció clasif_uni

```
function [ clasif ] = clasif_uni( M,u,f,d )
    s = size(M);%Dide dimensión de la matriz
    n = s(1);%Determina el número de filas de la matriz
    for i=1:n
        for j=1:n%Hace la matriz estocástica haciendo que el sumatorio de
las filas de cada columna sea la unidad
            v = sum(M(:,i));
            M1(j,i) = M(j,i)/v;
        end
    end
    P = (1-d)*(ones(n)*(1/n))+M1*d;%Crea la matriz estocástica P de
probabilidades
    [vapmax,vepmax] = vapvep(P);%Determina las matrices de valores propios
y vectores propios usando la función creada vapvep()
    for k1=1:u
        for k2 = 0:f-1%Crea un vector con los valores del vector propio
(puntuaciones) correspondientes a cada universidad para cada una de ellas
            uni(k2+1,1)=vepmax(k2*u+k1);
        end
        clasif(k1,1) = sum(uni);%Para cada universidad, suma el vector de
sus puntuaciones para obtener su puntuación final
    end
end
```